UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/883,302	06/19/2001	Jeffrey A. Bedell	53470.003004	9726

21967 7590 11/01/2010

HUNTON & WILLIAMS LLP
INTELLECTUAL PROPERTY DEPARTMENT
1900 K STREET, N.W.
SUITE 1200
WASHINGTON, DC 20006-1109

EXAMINER	
CHBOUKI, TAREK	

ART UNIT	PAPER NUMBER
2165	

MAIL DATE	DELIVERY MODE
11/01/2010	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 09/883,302	Applicant(s) BEDELL ET AL.	
	Examiner TAREK CHBOUKI	Art Unit 2165	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *04 October 2010*.
2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.
3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-24* is/are pending in the application.
4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5)☐ Claim(s) _____ is/are allowed.
6)☒ Claim(s) *1-24* is/are rejected.
7)☐ Claim(s) _____ is/are objected to.
8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.
10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a)☐ All b)☐ Some * c)☐ None of:
1.☐ Certified copies of the priority documents have been received.
2.☐ Certified copies of the priority documents have been received in Application No. _____.
3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Continued Examination Under 37 CFR 1.114

A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 10/04/2010 has been entered.

Response to Amendment

This Office action has been issued in response to amendment filed on 10/04/2010. Claims 1-24 are pending. Applicants' arguments have been carefully and respectfully considered and found not persuasive.

Response to arguments

With respect to Applicant's argument regarding the 35 USC 101. Examiner respectfully disagrees. Claims 1-9 are system claims and the bilski analysis is directed to method claims. Examiner acknowledge the amendment of claim 1-9 by introducing a computer processor in the preamble of the claims but still the step in the body of claims could be interpreted as software module within the computer device. Examiner suggests Applicant to incorporate the computer hardware (i.e processor or memory) in the body of the claims.

With respect to Applicant's argument on pages 12-13 and 15 stating that the 103(a) references do not teach or suggest "at least one query language statement having a tree query structure generated based at least in part on the parameters of the desired data set is assembled to be run against a data source to return the desired data set". Examiner respectfully disagrees.

Kobayashi references discloses in Fig. 11 and column 16, lines 63-67, column 18, lines 4-34, wherein query tree structure is created and wherein the parameter of the desired data is the input of the execution procedure). Therefore Kobayashi disclosure moots the above argument as claimed.

With respect to Applicant's argument on pages 13 and 16 stating that the 103(a) references do not teach or suggest "a statement assembly module for populating the syntax pattern, in the automated process with an argument data set associated with parameters of a desired data set and the desired function". Examiner respectfully disagrees. Firstly, Examiner would like to point out that the claimed desired function is so broad enabling one ordinary skill in to reasonably conclude the desire function is basically the query input. Therefore the parameter population of Saiki (Fig. 6 and paragraphs [0074], [0080] and [0081]) reads on the above mentioned argument.

With respect to Applicant's argument on page 14 stating that the 103(a) references do not teach or suggest and even teach away from "a syntax pattern selector module for selecting, in an automated process, a syntax pattern corresponding to a desired function and a syntax standards". Examiner respectfully disagrees. Firstly, As mentioned above, the claimed desired function is so broad enabling one ordinary skill in to reasonably conclude the desire function is basically the query input. Saeki in paragraph [0091], discloses the use of syntax standards. Secondly, with respect argument, Examiner is requesting Applicant to refer to the below mentioned fact about teaching away.

D. References Teach Away from the Invention or Render Prior Art Unsatisfactory for Intended Purpose

In addition to the material below, see **MPEP § 2141.02 <http://www.uspto.gov/web/offices/pac/mpep/documents/2100_2141_02.htm>** (prior art must be considered in its entirety, including disclosures that teach away from the claims) and **MPEP § 2143.01 <http://www.uspto.gov/web/offices/pac/mpep/documents/2100_2143_01.htm>** (proposed modification cannot render the prior art unsatisfactory for its intended purpose or change the principle of operation of a reference).

The Nature of the Teaching Is Highly Relevant

A prior art reference that "teaches away" from the claimed invention is a significant factor to be considered in determining obviousness; however, "the nature of the teaching is highly relevant and must

be weighed in substance. A known or obvious composition does not become patentable simply because it has been described as somewhat inferior to some other product for the same use." *In re Gurley*, 27 F.3d 551, 554, 31 USPQ2d 1130, 1132 (Fed. Cir. 1994) (Claims were directed to an epoxy resin based printed circuit material. A prior art reference disclosed a polyester-imide resin based printed circuit material, and taught that although epoxy resin based materials have acceptable stability and some degree of flexibility, they are inferior to polyester-imide resin based materials. The court held the claims would have been obvious over the prior art because the reference taught epoxy resin based material was useful for applicant's purpose, applicant did not distinguish the claimed epoxy from the prior art epoxy, and applicant asserted no discovery beyond what was known to the art.).

Furthermore, "the prior art's mere disclosure of more than one alternative does not constitute a teaching away from any of these alternatives because such disclosure does not criticize, discredit, or otherwise discourage the solution claimed.." *In re Fulton*, 391 F.3d 1195, 1201, 73 USPQ2d 1141, 1146 (Fed. Cir. 2004).

Proceeding Contrary to Accepted Wisdom Is Evidence of Nonobviousness

The totality of the prior art must be considered, and proceeding contrary to accepted wisdom in the art is evidence of nonobviousness. *In re Hedges,* 783 F.2d 1038, 228 USPQ 685 (Fed. Cir. 1986) (Applicant's claimed process for sulfonating diphenyl sulfone at a temperature above 127°C was contrary to accepted wisdom because the prior art as a whole suggested using lower temperatures for optimum results as evidenced by charring, decomposition, or reduced yields at higher temperatures.).

Furthermore, "[k]nown disadvantages in old devices which would naturally discourage search for new inventions may be taken into account in determining obviousness." *United States v. Adams,* 383 U.S. 39, 52, 148 USPQ 479, 484 (1966).

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:
Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

1. Claims 1-9 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claims 1-9 refer to a “system”. As cited on page 2 of this instant specification, has provided evidence that the claimed system is a software per se, wherein a series of modules are to be executed. The

claims do not define structural and functional descriptive material used in interrelationship between the computer software and the hardware like a memory or processor.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1-3, 8-9 and 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Saeki, Joji (hereinafter Saeki) US Publication No 20040039730 and Stern, Jonathan et al (hereinafter Stern) US Publication No. 20020032740 in view of Kobayashi, Susumu et al (hereinafter Kobayashi) US Patent No. 6212516.

As per claim 1, Saeki discloses:

A <u>computer-implemented</u> system <u>comprises at least a programmed computer processor</u> for automated generation of one or more query language statements, <u>the computer implemented system</u> comprising:

a syntax pattern selector module for selecting a syntax pattern corresponding to a desired function defining a default syntax pattern provided to the syntax pattern selector module and a syntax standard for use in generating the one or more query language statements;

(Fig. 4-5 and paragraphs [0080]-[0081] and 0091], indicate the selection of definition of a language syntax pattern to replace variable and automatically generates a query statement and wherein the syntax definition contains default reference establishing data join (default syntax pattern))

a statement assembly module for populating the syntax pattern with an argument data set associated with parameters of a desired data set provided to the statement assembly module as part of the process of generating the one or more query language statements;

(Fig. 6 and paragraphs [0074], [0080] and [0081], wherein the replacement of variable is the population of syntax and wherein parameters (references) control a retrieval request designation (desired data)).

and whereby at least one query language statement is assembled to be run against a data source to return the desired data set.

(Paragraph [0066]).

Saeki does not go into detail regarding the selection mechanism being automated, however in an analogous art of data pattern matching, Stern teaches:

syntax pattern selector module for selecting, in the automated process

(Paragraph [0116], wherein syntax pattern is automated and selected based on template)

Therefore, it would have been obvious to a person in the ordinary skill in the art at the time of the invention to combine Saeki with Stern by incorporating the teaching of Stern into the method of Saeki. One having ordinary skill in the art would have found it motivated to use the automation process of Stern into the system of Saeki for the purpose of enabling pattern recognition to take place without user intervention.

Saeki and Stern do not go into detail regarding creating tree query structure, however in an analogous art of data pattern matching, Kobayashi teaches:

at least one query language statement having a tree query structure <u>generated based at least in part on the parameters of the desired dataset</u>

(Fig. 11 and column 16, lines 63-67, column 18, lines 4-34, wherein query tree structure is created and wherein the parameter of the desired data is the input of the execution procedure). Therefore, it would have been obvious to a person in the ordinary skill in the art at the time of the invention to combine Saeki with Stern and Kobayashi by incorporating the teaching of Kobayashi into the method of Saeki and Stern. One having ordinary skill in the art would have found it motivated to use the tree query structure of Kobayashi into the system of Saeki and Stern for the purpose of leveraging tree structure usage/creation for querying plurality of data source or monitoring execution plan.

As per claim 2, Saeki and Stern and Kobayashi teach:

The <u>computer-implemented</u> system of claim 1, wherein the syntax pattern selector module selects the syntax pattern from a plurality of syntax patterns corresponding to a plurality of database management systems.

(Paragraph [0033] and [0069] and [0133], indicate syntax analysis (selection) of the syntax patterns corresponding to multiple databases definitions)(Saeki)

As per claim 3, Saeki and Stern and Kobayashi teach:

The <u>computer-implemented</u> system of claim 1, wherein the syntax pattern selector module selects the syntax pattern from a plurality of syntax patterns based upon at least one selection variable.

(Fig. 6 and paragraph [0080], indicate including a variable to establish the pattern)(Saeki)

As per claim 8, Saeki and Stern and Kobayashi teach:

The <u>computer-implemented</u> system of claim 1, wherein the system is a component in an online analytical processing system, a reporting system, a business intelligence system, or a data mining system.

(Paragraph [0018], wherein the system is online analytical processing system)(Saeki).

As per claim 9 Saeki and Stern and Kobayashi teach:

The <u>computer-implemented</u> system of claim 1, further comprising a plurality of driver modules,

each of the driver modules including at least one syntax pattern associated with a selected database management system.

(Paragraph [0033] and [0069] and [0075] and [0133], indicate syntax analysis (selection) of the syntax patterns corresponding to multiple databases definitions and wherein the logical item type definition is the driver module used to define or update syntax pattern)(Saeki)

As per claim 20, Saeki discloses:

A tangible medium having a computer readable program code embodied therein for generating one or more query language statements through an computer-implemented method comprising:

code for causing the processor to identify a functional element corresponding to a desired function for use in generating the one or more query language statements, wherein the functional element defines a default syntax pattern;

(Paragraphs [0033] and [0080]-[0081] and [0091], Wherein the definition (identifier) and syntax analysis corresponding retrieval request and generated query syntax and wherein the syntax definition contains default reference establishing data join (default syntax pattern))

code for causing the processor to identify an argument data set associated with parameters

of a desired data set and the identified functional element as part of generating the one or more query language statements;

(Fig. 6 and paragraphs [0074], [0080] and [0081], wherein the replacement of variable is the identification of syntax argument and wherein parameters (references) control a retrieval request designation (desired data)).

code for causing the processor to select a syntax pattern corresponding to the functional element;

(Paragraph [0033] and [0080], Wherein the definition (identifier) and syntax analysis corresponding retrieval request and generated query syntax).

and code for causing the processor to populate the selected syntax pattern with the identified argument data set to assemble at least one query language statement to be run against a data source to return the desired data set.

(Paragraphs [0066] and [0080] and [0081], wherein the replacement of variable is "augmentation of data" used during query statement generation to be executed against a database and retrieve desired information).

Saeki does not go into detail regarding the selection mechanism being automated, however in an analogous art of data pattern matching, Stern teaches:

generating one or more query language statement in an automated process

(Paragraph [0116], wherein syntax pattern is automated and selected based on template)

Therefore, it would have been obvious to a person in the ordinary skill in the art at the time of the invention to combine Saeki with Stern by incorporating the teaching of Stern into the method of Saeki. One having ordinary skill in the art would have found it motivated to use the automation process of Stern into the system of Saeki for the purpose of enabling pattern recognition to take place without user intervention.

Saeki and Stern do not go into detail regarding creating tree query structure, however in an analogous art of data pattern matching, Kobayashi teaches:

at least one query language statement having a tree query structure <u>generated based at least in part on the parameters of the desired dataset</u>

(Fig. 11 and column 16, lines 63-67, column 18, lines 4-34, wherein query tree structure is created and wherein the parameter of the desired data is the input of the execution procedure).

Therefore, it would have been obvious to a person in the ordinary skill in the art at the time of the invention to combine Saeki with Stern and Kobayashi by incorporating the teaching of Kobayashi into the method of Saeki and Stern. One having ordinary skill in the art would have found it motivated to use the tree query structure of Kobayashi into the system of Saeki and Stern for the purpose of leveraging tree structure usage/creation for querying plurality of data source or monitoring execution

3. Claims 4-7, 10-19 and 21-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Saeki, Joji (hereinafter Saeki) US Publication No 2004/0039730 in view of Kobayashi, Susumu et al (hereinafter Kobayashi) US Patent No. 6212516.

As per claim 4, Saeki discloses:

A <u>computer-implemented</u> system <u>comprises at least a programmed computer processor</u> for generating one or more query language statements, the <u>computer-implemented</u> system comprising:

a syntax pattern selector module for selecting a syntax pattern corresponding to a desired function defining a default syntax pattern, and a syntax standard;

(Fig. 4-5 and paragraphs [0080]-[0081] and [0091], indicate the selection of definition of a language syntax pattern to replace variable and automatically generates a query statement and wherein the syntax definition contains default reference establishing data join (default syntax pattern))

a statement assembly module for populating the syntax pattern with an argument data set associated with parameters of a desired data set;

(Fig. 6 and paragraphs [0074], [0080] and [0081], wherein the replacement of variable is the population of syntax and wherein parameters (references) control a retrieval request designation (desired data)).

a structure generator module for generating a query structure based on the desired data set, the query structure providing a basis for identifying the desired function to be used by the syntax pattern selector module;

(Fig. 4-5 and paragraph [0080], indicate the selection of definition of a language syntax pattern to replace variable and automatically generates a query statement)

and whereby at least one query language statement is assembled to be run against a data source to return the desired data set.

(Paragraph [0066]).

Saeki does not go into detail regarding creating tree query selection, however in an analogous art of data pattern matching, Kobayashi teaches:

at least one query language statement having a tree query structure <u>generated based at least in part on the parameters of the desired dataset</u>

(Fig. 11 and column 16, lines 63-67, column 18, lines 4-34, wherein query tree structure is created and wherein the parameter of the desired data is the input of the execution procedure).

Therefore, it would have been obvious to a person in the ordinary skill in the art at the time of the invention to combine Saeki and Kobayashi by incorporating the teaching of Kobayashi into the method of Saeki. One having ordinary skill in the art would have found it motivated to use the tree query structure of Kobayashi into the system of Saeki for the purpose of leveraging tree structure usage/creation for querying plurality of data source or monitoring execution plan.

As per claim 5, Saeki discloses:

A <u>computer-implemented</u> system <u>comprises at least a programmed computer processor</u> for generating one or more query language statements, <u>computer-implemented</u> system comprising:

a syntax pattern selector module for selecting a syntax pattern corresponding to a desired function defining a default syntax pattern, and a syntax standard;

(Fig. 4-5 and paragraphs [0080]-[0081] and [0091], indicate the selection of definition of a language syntax pattern to replace variable and automatically generates a query statement and wherein the syntax definition contains default reference establishing data join (default syntax pattern))

a statement assembly module for populating the syntax pattern with an argument data set associated with parameters of a desired data set <u>and the desired function</u>;

(Fig. 6 and paragraphs [0074], [0080] and [0081], wherein the replacement of variable is the population of syntax and wherein parameters (references) control a retrieval request designation (desired data)).

a function identifier module for identifying a functional element corresponding to the desired function and at least one syntax pattern;

(Paragraph [0033] and [0080], Wherein the definition (identifier) corresponding retrieval request and generated query syntax).

and whereby at least one query language statement is assembled to be run against a data source to return the desired data set.

(Paragraph [0066]).

Saeki does not go into detail regarding creating tree query selection, however in an analogous art of data pattern matching, Kobayashi teaches:

at least one query language statement having a tree query structure <u>generated based at least in part on the parameters of the desired dataset</u>

(Fig. 11 and column 16, lines 63-67, column 18, lines 4-34, wherein query tree structure is created and wherein the parameter of the desired data is the input of the execution procedure).

Therefore, it would have been obvious to a person in the ordinary skill in the art at the time of the invention to combine Saeki and Kobayashi by incorporating the teaching of Kobayashi into the method of Saeki. One having ordinary skill in the art would have found it motivated to use the tree query structure of Kobayashi into the system of Saeki for the purpose of leveraging tree structure usage/creation for querying plurality of data source or monitoring execution plan.

As per claim 6, Saeki discloses:

A <u>computer-implemented</u> system <u>comprises at least a programmed computer processor</u> for generating one or more query language statements, <u>computer-implemented</u> system comprising:

a syntax pattern selector module for selecting a syntax pattern corresponding to a desired function defining a default syntax pattern, and a syntax standard;

(Fig. 4-5 and paragraphs [0080]-[0081] and [0091], indicate the selection of definition of a language syntax pattern to replace variable and automatically generates a query statement and wherein the syntax definition contains default reference establishing data join (default syntax pattern))

a statement assembly module for populating the syntax pattern with an argument data set associated with parameters of a desired data set <u>and the desired function</u>;

(Fig. 6 and paragraphs [0074], [0080] and [0081], wherein the replacement of variable is the population of syntax and wherein parameters (references) control a retrieval request designation (desired data)).

an argument generator module for identifying the argument data set associated with the desired data set;

(Fig. 6 and paragraph [0080] and [0081], wherein the replacement of variable is “augmentation of data” incorporated identifying the variable (dataset))

and whereby at least one query language statement is assembled to be run against a data source to return the desired data set.

(Paragraph [0066]).

Saeki does not go into detail regarding creating tree query selection, however in an analogous art of data pattern matching, Kobayashi teaches:

at least one query language statement having a tree query structure <u>generated based at least in part on the parameters of the desired dataset</u>

(Fig. 11 and column 16, lines 63-67, column 18, lines 4-34, wherein query tree structure is created and wherein the parameter of the desired data is the input of the execution procedure).

Therefore, it would have been obvious to a person in the ordinary skill in the art at the time of the invention to combine Saeki and Kobayashi by incorporating the teaching of Kobayashi into the method of Saeki. One having ordinary skill in the art would have found it motivated to use the tree query structure of Kobayashi into the system of Saeki for the purpose of leveraging tree structure usage/creation for querying plurality of data source or monitoring execution plan.

As per claim 7, Saeki and Kobayashi teach:

The <u>computer-implemented</u> system of claim 6, wherein the argument generator module identifies

the argument data set based upon a syntax description associated with the desired function.

(Fig. 6 and paragraph [0080] and [0081], wherein the replacement of variable is "augmentation of data" using definition information of a language syntax)(Saeki)

As per claim 10, Saeki discloses:

A computer-implemented method of generating one or more query language statements to be run against a one or more data sources, comprising the steps, performed by a computer system, of:

selecting a syntax pattern corresponding to a desired function defining a default syntax pattern, provided as an input and a syntax standard for use in generating the one or more query language statements;

(Fig. 4-5 and paragraphs [0080]-[0081] and [0091], indicate the selection of definition of a language syntax pattern to replace variable and automatically generates a query statement and wherein the syntax definition contains default reference establishing data join (default syntax pattern))

populating the syntax pattern with an argument data set associated with parameters of a desired data set <u>and the desired function</u> provided as an input identifying the data set on which to operate from the data source as part of generating the one or more query language statements;

(Fig. 6 and paragraphs [0074], [0080] and [0081], wherein the replacement of variable is the population of syntax and wherein parameters (references) control a retrieval request designation (desired data)).

and wherein, the populated syntax pattern is used to generate one or more query language statements runnable against one or more data sources to return the desired data result set.

(Paragraph [0066]).

Saeki does not go into detail regarding creating tree query selection, however in an analogous art of data pattern matching, Kobayashi teaches:

at least one query language statement having a tree query structure <u>generated based at least in part on the parameters of the desired dataset</u>

(Fig. 11 and column 16, lines 63-67, column 18, lines 4-34, wherein query tree structure is created and wherein the parameter of the desired data is the input of the execution procedure).

Therefore, it would have been obvious to a person in the ordinary skill in the art at the time of the invention to combine Saeki and Kobayashi by incorporating the teaching of Kobayashi into the method of

Saeki. One having ordinary skill in the art would have found it motivated to use the tree query structure of Kobayashi into the system of Saeki for the purpose of leveraging tree structure usage/creation for querying plurality of data source or monitoring execution plan.

As per claim 11, Saeki and Kobayashi teach:

The <u>computer-implemented</u> method of claim 10, wherein the step of selecting the syntax pattern includes selecting the syntax pattern from a plurality of syntax patterns corresponding to a plurality of database management systems.

(Paragraph [0033] and [0069] and [0133], indicate syntax analysis (selection) of the syntax patterns corresponding to multiple databases definitions)(Saeki)

As per claim 12, Saeki and Kobayashi teach:

The <u>computer-implemented</u> method of claim 10, wherein the step of selecting the syntax pattern includes selecting the syntax pattern from a plurality of syntax patterns based upon at least one selection variable.

(Fig. 6 and paragraph [0080], indicate including a variable to establish the pattern)(Saeki)

As per claim 13, Saeki and Kobayashi teach:

The <u>computer-implemented</u> method of claim 10, further comprising the step of generating a query structure based on the desired data set, the query structure providing a basis for identifying the desired function to be used in selecting the syntax pattern.

(Fig. 4-5 and paragraph [0080], indicate the selection of definition of a language syntax pattern to replace variable and automatically generates a query statement)(Saeki)

As per claim 14, Saeki discloses:

A method of generating one or more query language statements to be run against a one or more data sources, comprising the steps of:

selecting a syntax pattern corresponding to a desired function defining a default syntax pattern, and a syntax standard;

(Fig. 4-5 and paragraphs [0080]-[0081] and [0091], indicate the selection of definition of a language syntax pattern to replace variable and automatically generates a query statement and wherein the syntax definition contains default reference establishing data join (default syntax pattern))

populating the syntax pattern with an argument data set associated with parameters of a desired data set from the data source <u>and the desired function</u>;

(Fig. 6 and paragraphs [0074], [0080] and [0081], wherein the replacement of variable is the population of syntax and wherein parameters (references) control a retrieval request designation (desired data)).

identifying a functional element corresponding to the desired function and at least one syntax pattern, the functional element providing a basis for selecting the syntax pattern;

(Paragraph [0033] and [0080], Wherein the definition (identifier) corresponding retrieval request and generated query syntax).

and wherein, the populated syntax pattern comprises one or more query language statements runnable against one or more data sources to return the desired data result set.

(Paragraph [0066]).

Saeki does not go into detail regarding creating tree query selection, however in an analogous art of data pattern matching, Kobayashi teaches:

at least one query language statement having a tree query structure <u>generated based at least in part on the parameters of the desired dataset</u>

(Fig. 11 and column 16, lines 63-67, column 18, lines 4-34, wherein query tree structure is created and wherein the parameter of the desired data is the input of the execution procedure).

Therefore, it would have been obvious to a person in the ordinary skill in the art at the time of the invention to combine Saeki and Kobayashi by incorporating the teaching of Kobayashi into the method of Saeki. One having ordinary skill in the art would have found it motivated to use the tree query structure of Kobayashi into the system of Saeki for the purpose of leveraging tree structure usage/creation for querying plurality of data source or monitoring execution plan.

As per claim 15, Saeki and Kobayashi teach:

The <u>computer-implemented</u> method of claim 10, further comprising the step of identifying at least one selection variable for selecting the syntax pattern from a plurality of syntax patterns.

(Paragraph [0033] and [0069] and [0133], indicate syntax analysis (selection) of the syntax patterns corresponding to multiple databases definitions)(Saeki)

As per claim 16, Saeki discloses:

A method of generating one or more query language statements to be run against a one or more data sources, comprising the steps of:

selecting a syntax pattern corresponding to a desired function defining a default syntax pattern, and a syntax standard;

(Fig. 4-5 and paragraphs [0080]-[0081] and [0091], indicate the selection of definition of a language syntax pattern to replace variable and automatically generates a query statement and wherein the syntax definition contains default reference establishing data join (default syntax pattern))

populating the syntax pattern with an argument data set associated with a desired data set from the data source <u>and the desired function</u>;

(Fig. 6 and paragraph [0080] and [0081], wherein the replacement of variable is "augmentation of data")

identifying the argument data set associated with parameters of the desired data set;

(Fig. 6 and paragraphs [0074], [0080] and [0081], wherein the replacement of variable is the identification of syntax argument and wherein parameters (references) control a retrieval request designation (desired data)).

and wherein, the populated syntax pattern comprises one or more query language statements runnable against one or more data sources to return the desired data result set.

(Paragraph [0066]).

Saeki does not go into detail regarding creating tree query selection, however in an analogous art of data pattern matching, Kobayashi teaches:

at least one query language statement having a tree query structure <u>generated based at least in part on the parameters of the desired dataset</u>

(Fig. 11 and column 16, lines 63-67, column 18, lines 4-34, wherein query tree structure is created and wherein the parameter of the desired data is the input of the execution procedure).

Therefore, it would have been obvious to a person in the ordinary skill in the art at the time of the invention to combine Saeki and Kobayashi by incorporating the teaching of Kobayashi into the method of Saeki. One having ordinary skill in the art would have found it motivated to use the tree query structure of Kobayashi into the system of Saeki for the purpose of leveraging tree structure usage/creation for querying plurality of data source or monitoring execution plan.

As per claim 17, Saeki and Kobayashi teach:

The method of claim 16, wherein the step of identifying the argument data set includes identifying the argument data set based upon a syntax description associated with the desired function.

(Fig. 6 and paragraph [0080] and [0081], wherein the replacement of variable is "augmentation of data" incorporated identifying the variable (dataset))(Saeki)

As per claim 18, Saeki and Kobayashi teach:

The method of claim 10, wherein the method is executed in an online analytical processing systems, a reporting system, a business intelligence system, or a data mining system.

(Paragraph [0018], wherein the system is online analytical processing system)(Saeki).

As per claim 19, Saeki and Kobayashi teach:

The method of claim 10, wherein the step of selecting the syntax pattern includes accessing a plurality of driver modules including at least one syntax pattern, each of the plurality of driver modules corresponding to a selected database management system.

(Paragraph [0033] and [0069] and [0075] and [0133], indicate syntax analysis (selection) of the syntax patterns corresponding to multiple databases definitions and wherein the logical item type definition is the driver module used to define or update syntax pattern)(Saeki) .

As per claim 21, Saeki discloses:

A method of generating a query language statement from computer code embodied on a computer readable media comprising the steps of:

defining a syntax pattern accessible to a system for generating a query language statement;

(Fig. 4-5 and paragraph [0080], indicate the selection of definition of a language syntax pattern to replace variable and automatically generates a query statement)

accessing the defined syntax pattern from the system to generate a query language statement;

(Fig. 4-5 and paragraph [0080], indicate the selection of definition of a language syntax pattern to replace variable and automatically generates a query statement)

and wherein the system does not need to be recompiled as a result of defining the syntax pattern.

(Fig. 4-5 and paragraph [0080], indicate the selection of definition of a language syntax pattern to replace variable and automatically generates a query statement)

Saeki does not go into detail regarding creating tree query selection, however in an analogous art of data pattern matching, Kobayashi teaches:

at least one query language statement having a tree query structure <u>generated based at least in part on the parameters of the desired dataset</u>

(Fig. 11 and column 16, lines 63-67, column 18, lines 4-34, wherein query tree structure is created and wherein the parameter of the desired data is the input of the execution procedure).

Therefore, it would have been obvious to a person in the ordinary skill in the art at the time of the invention to combine Saeki and Kobayashi by incorporating the teaching of Kobayashi into the method of Saeki. One having ordinary skill in the art would have found it motivated to use the tree query structure of Kobayashi into the system of Saeki for the purpose of leveraging tree structure usage/creation for querying plurality of data source or monitoring execution plan.

As per claim 22, Saeki and Kobayashi teach:

The method of claim 21, wherein the syntax pattern is associated with a selected database management system.

(Paragraph [0033] and [0069] and [0133], indicate syntax analysis (selection) of the syntax patterns corresponding to multiple databases definitions)(Saeki)

As per claim 23, Saeki discloses:

A tangible medium having a computer readable program code embodied therein for generating one or more query language statements comprising:

code for causing the processor to identify a functional element corresponding to a desired function, wherein the functional element defines a default syntax pattern;

(Paragraphs [0033] and [0080]-[0081], Wherein the definition (identifier) and syntax analysis corresponding retrieval request and generated query syntax and wherein the syntax definition contains default reference establishing data join (default syntax pattern))

code for causing the processor to identify an argument data set associated with parameters of a desired data set and the identified functional element;

(Fig. 6 and paragraphs [0074], [0080] and [0081], wherein the replacement of variable is the identification of syntax argument and wherein parameters (references) control a retrieval request designation (desired data)).

code for causing the processor to select a syntax pattern corresponding to the functional element;

(Paragraph [0033] and [0080], Wherein the definition (identifier) corresponding retrieval request and generated query syntax).

code for identifying a functional element corresponding to the desired function and at least one syntax pattern, the functional element providing a basis for selecting the syntax pattern;

(Paragraph [0033] and [0080], Wherein the definition (identifier) and syntax analysis corresponding retrieval request and generated query syntax).

and code for causing the processor to populate the selected syntax pattern with the identified argument data set to assemble at least one query language statement to be run against a data source to return the desired data set.

(Paragraphs [0066] and [0080] and [0081], wherein the replacement of variable is "augmentation of data" used during query statement generation to be executed against a database and retrieve desired information).

Saeki does not go into detail regarding creating tree query selection, however in an analogous art of data pattern matching, Kobayashi teaches:

at least one query language statement having a tree query structure <u>generated based at least in part on the parameters of the desired dataset</u>

(Fig. 11 and column 16, lines 63-67, column 18, lines 4-34, wherein query tree structure is created and wherein the parameter of the desired data is the input of the execution procedure).

Therefore, it would have been obvious to a person in the ordinary skill in the art at the time of the invention to combine Saeki and Kobayashi by incorporating the teaching of Kobayashi into the method of Saeki. One having ordinary skill in the art would have found it motivated to use the tree query structure of Kobayashi into the system of Saeki for the purpose of leveraging tree structure usage/creation for querying plurality of data source or monitoring execution plan.

As per claim 24, Saeki discloses:

A tangible medium having a computer readable program code embodied therein for generating one or more query language statements comprising:

code for causing the processor to identify a functional element corresponding to a desired function, wherein the functional element defines a default syntax pattern;

(Paragraph [0033] and [0080], Wherein the definition (identifier) and syntax analysis corresponding retrieval request and generated query syntax).

code for causing the processor to identify an argument data set associated with a desired data set and the identified functional element;

(Fig. 6 and paragraph [0080] and [0081], wherein the definition syntax information manages the replacement of variable is "augmentation of data" incorporated identifying the variable (dataset))

code for causing the processor to select a syntax pattern corresponding to the functional element;

(Paragraph [0033] and [0080], Wherein the definition (identifier) and syntax analysis corresponding retrieval request and generated query syntax).

code for identifying the argument data set associated with parameters of the desired data set;

(Fig. 6 and paragraphs [0074], [0080] and [0081], wherein the replacement of variable is the identification of syntax argument and wherein parameters (references) control a retrieval request designation (desired data)).

and code for causing the processor to populate the selected syntax pattern with the identified argument data set to assemble at least one query language statement to be run against a data source to return the desired data set.

(Paragraphs [0066] and [0080] and [0081], wherein the replacement of variable is "augmentation of data" used during query statement generation to be executed against a database and retrieve desired information).

Saeki does not go into detail regarding creating tree query selection, however in an analogous art of data pattern matching, Kobayashi teaches:

at least one query language statement having a tree query structure <u>generated based at least in part on the parameters of the desired dataset</u>

(Fig. 11 and column 16, lines 63-67, column 18, lines 4-34, wherein query tree structure is created and wherein the parameter of the desired data is the input of the execution procedure).

Therefore, it would have been obvious to a person in the ordinary skill in the art at the time of the invention to combine Saeki and Kobayashi by incorporating the teaching of Kobayashi into the method of Saeki. One having ordinary skill in the art would have found it motivated to use the tree query structure of Kobayashi into the system of Saeki for the purpose of leveraging tree structure usage/creation for querying plurality of data source or monitoring execution plan.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tarek Chbouki whose telephone number is 571-2703154. The examiner can normally be reached on Mon-Fri 7:30 am to 5:00 pm EST. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Neveen Abel-Jalil can be reached at 571-2724074. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/TAREK CHBOUKI/

Examiner, Art Unit 2165

10/30/2010